Unit 2 Seminar

Title: Scrum Security Review

Question 1: Table

Create a 2-column multi-line table. In the left-hand column, include the software development stages of the Scrum agile life cycle approach to project management. In the right-hand column, describe the processes you recommend applied at each stage to ensure that secure software is produced at the end of the development. To support the preparation of your response, you can refer to the following literature:

Sharma, A. & Bawa, R. K. (2020) Identification and Integration of Security Activities for Secure Agile Development. *International Journal of Information Technology*.

My answer:

Sharma & Bawa (2020) state that Agile development strongly emphasises informal, dynamic, and implicit knowledge-driven approaches to produce high business-value projects, with little formalisation needed where essential. These principles are outlined in the Agile Manifesto, which also emphasises the importance of early and continuous software delivery to satisfy customers, the acceptance of changing requirements, and the use of working software to gauge progress. The most remarkable ideas and architectures can evolve thanks to self-organising teams. However, some security-related drawbacks exist because of incompatibilities with traditional safe software development techniques.

Depending on the project management methodology, the Agile life cycle varies. Sprints, which are brief work intervals for Scrum teams, are determined by roles like

the Scrum master. Teams using Kanban operate in a continuous flow with no set roles. Extreme Programming teams concentrate on engineering practices and work in shorter iterations. All software development teams, however, strive to provide users with functional software on schedule (Wrike, 2022).

Scrum is an agile project management framework that uses values, principles, and practices to help teams structure and manage their work. It encourages self-organisation, learning through experiences, and continuous improvement. It can be applied to various types of teamwork and includes meetings, tools, and roles to support the process (Drumond, 2018).

Based on the article by Drummond (2018), the main steps comprise the following phases: Product Backlog, Sprint Planning, Sprint Execution, Sprint Review, and Sprint Retrospective.

1. Product Backlog refinement involves breaking down more oversized items into smaller, specific ones. It can be done formally or through ongoing discussions during a Sprint. Refinement can increase transparency and accuracy (Scrum, N.D.). Requirements gathering understands what is needed to build and why, whether for software or any other project, and the process involves gathering requirements from stakeholders, documenting them as user stories and feature specs, and ensuring everyone understands the project goals (Hirsch, 2017).
2. Product Backlog, as Shechter (N.D) defines, is the stage in which user stories and requirements are created and prioritised in a product backlog. According to Atlassian (2023), the *backlog* is establishing acceptability standards and security specifications for each user narrative. Based on risk analysis, prioritise user stories relevant to security and review the product's security architecture and threat modelling.
3. Sprint planning entails choosing the user stories for the forthcoming sprint and task planning (Visual-Paradigm (N.D.). For instance, SAFECode (2012) guides the development team's responsibility for security-related duties, determines the time and effort needed for security activities, and specifies the sprint's security requirements and restrictions. Sprint planning involves the product owner defining the goal, the development team planning the work, and both parties negotiating based on value and effort. The product backlog and capacity are essential inputs, and the outcome is a visible sprint backlog with a clear goal (West, N.D.).

4. Sprint Execution: At this stage, the selected user stories are executed during the sprint execution (Knowledgehut (N.D.). According to Conklin & Robinson (2017), secure coding standards and recommendations should be followed, static code analysis and code reviews should be conducted to identify security flaws, and safe frameworks and libraries should be used to avoid common security vulnerabilities.
5. The sprint review gathers stakeholder feedback on completed user stories (Onuta, 2021). This stage involves showing stakeholders the product's security features and capabilities, gathering input and ideas for security enhancements, and updating the product backlog with new or updated security user stories (Scrum.org, N.D.).
6. Sprint Retrospective Retrospectives improve future outcomes by reflecting on the past (Atlassian, N.D.). In this phase, the security practices used in the sprint will be assessed for effectiveness and efficiency. Security issues and lessons gained from the sprint will also be identified, and security best practices and action items will be suggested for the following sprint (Atlassian, N.D.).

Below is a table that shows the stages of software development in the Scrum agile life cycle approach to project management:

Software development stages of the Scrum agile	**Recommended Security Processes**
Refining the Product Backlog: understanding what is needed to build and why, whether for software or any other project.	- Gathering requirements from stakeholders - Documenting them as user stories - Making sure that everyone comprehends the project objectives and specifications.
Product Backlog: User stories and requirements are created and prioritised	- Establishing acceptability standards and security specifications for each user narrative; - Based on risk analysis, prioritise user stories relevant to security; - and review the product's security architecture and threat modelling.
Sprint Planning: Selecting user stories for the upcoming sprint and planning tasks.	- Give the development team responsibility for security-related duties; - determines the time and effort needed for security activities; - and specifies the sprint's security requirements and restrictions; - Product owner defining the goal; - Development team planning the work; - Both parties negotiate based on value and effort.
Sprint Execution: User stories are executed	- Secure coding standards and recommendations should be followed;

	- Static code analysis and code reviews should be conducted to identify security flaws; - Safe frameworks and libraries should be used to avoid common security vulnerabilities.
Sprint Review: Showing stakeholders finished user stories.	- Showing stakeholders the product's security features and capabilities, - Gathering input and ideas for security enhancements, - and updating the product backlog with new or updated security user stories
Sprint Retrospective: Retrospectives enhance results in the future.	- Security practices used in the sprint will be assessed for effectiveness and efficiency; - Security issues and lessons gained from the sprint will also be identified, - Security best practices and action items will be suggested for the following sprint

References:

Sharma, A. & Bawa, R.K. (2020). Identification and integration of security activities for secure agile development. *International Journal of Information Technology*. doi:https://doi.org/10.1007/s41870-020-00446-4.

Wrike (2022). *The Agile Software Development Life Cycle | Wrike Agile Guide*. [online] Wrike. Available at: https://www.wrike.com/agile-guide/agile-development-life-cycle/.

Drumond, C. (2018). *Scrum - what it is, how it works, and why it's awesome*. [online] Atlassian. Available at: https://www.atlassian.com/agile/scrum.

Scrum (N.D.). *What is a Product Backlog?* [online] Available at: https://www.scrum.org/resources/what-is-a-product-backlog#:~:text=Product%20Backlog%20Refinement- [Accessed 29 Aug. 2023].

Hirsch, J. (2017). *10 Steps To Successful Requirements Gathering*. [online] Phase2. Available at: https://www.phase2technology.com/blog/successful-requirements-gathering.

Shechter, O. (N.D). *The Product Backlog: A Step-by-step Guide*. [online] Available at: https://www.toptal.com/product-managers/agile/product-backlog-step-by-step-guide.

Atlassian (2023). *Backlog Refinement Guide: How to & tips to be successful*. [online] Atlassian. Available at: https://www.atlassian.com/agile/scrum/backlog-refinement [Accessed 21 Aug. 2023].

Visual-Paradigm (N.D.). *How to Conduct a Sprint Planning Meeting*. [online] Available at: https://www.visual-paradigm.com/tutorials/agile-tutorial/how-to-conduct-

sprint-planning-meeting/#:~:text=During%20the%20sprint%20planning%20meeting [Accessed 21 Aug. 2023].

SAFECode (2012). *Practical Security Stories and Security Tasks for Agile Development Environments.* Available at: https://safecode.org/publication/SAFECode_Agile_Dev_Security0712.pdf.

West, D. (N.D.). *Sprint Planning*. [online] Atlassian. Available at: https://www.atlassian.com/agile/scrum/sprint-planning.

Knowledgehut (N.D.). *Sprint Execution Overview* | *Sprint Execution Activities*. [online] Available at: https://www.knowledgehut.com/tutorials/scrum-tutorial/sprint-execution.

Conklin, L. & Robinson, G. (2017). *CODE REVIEW GUIDE RELEASE Creative Commons (CC) Attribution*. [online] Available at: https://owasp.org/www-pdf-archive/OWASP_Code_Review_Guide_v2.pdf.

Onuta, A. (2021). *12 Things you Must Know About the Sprint Review*. [online] Medium. Available at: https://ancaonuta.medium.com/12-things-you-must-know-about-the-sprint-review-e57cfea4da3d#:~:text=Demonstrate%20the%20work%20done%20%E2%80%94%20take [Accessed 21 Aug. 2023].

Scrum.org. (n.d.). *What is a Sprint Review?* [online] Available at: https://www.scrum.org/resources/what-is-a-sprint-review.

Atlassian (N.D). *Agile retrospectives: Use the past to define the future*. [online] Atlassian. Available at: https://www.atlassian.com/agile/scrum/retrospectives.